Apollo
Roman
Adam

8-Bit Or Gate

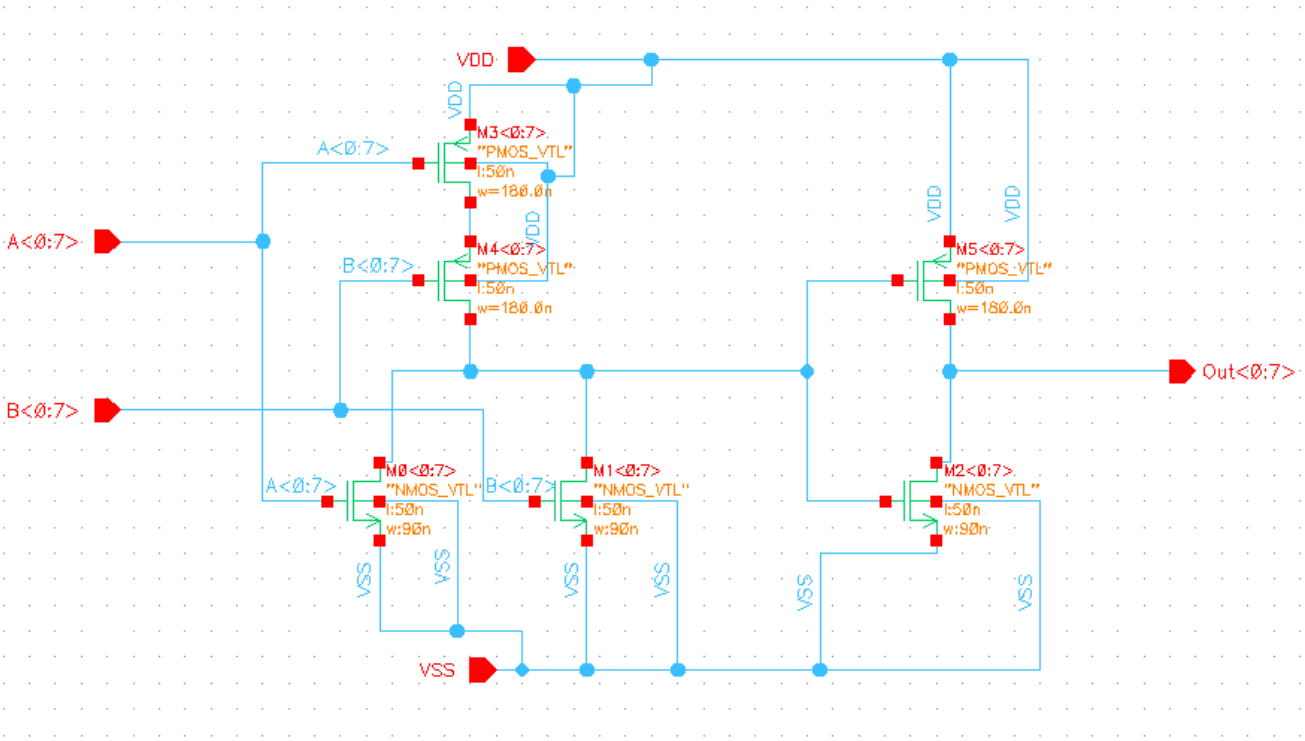

Figure 1: Schematic View of Or Gate

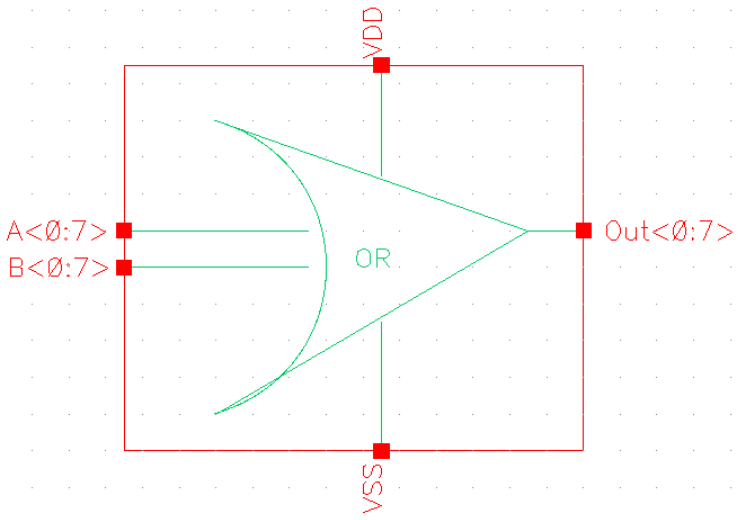

Figure 2: Symbol View of Or Gate

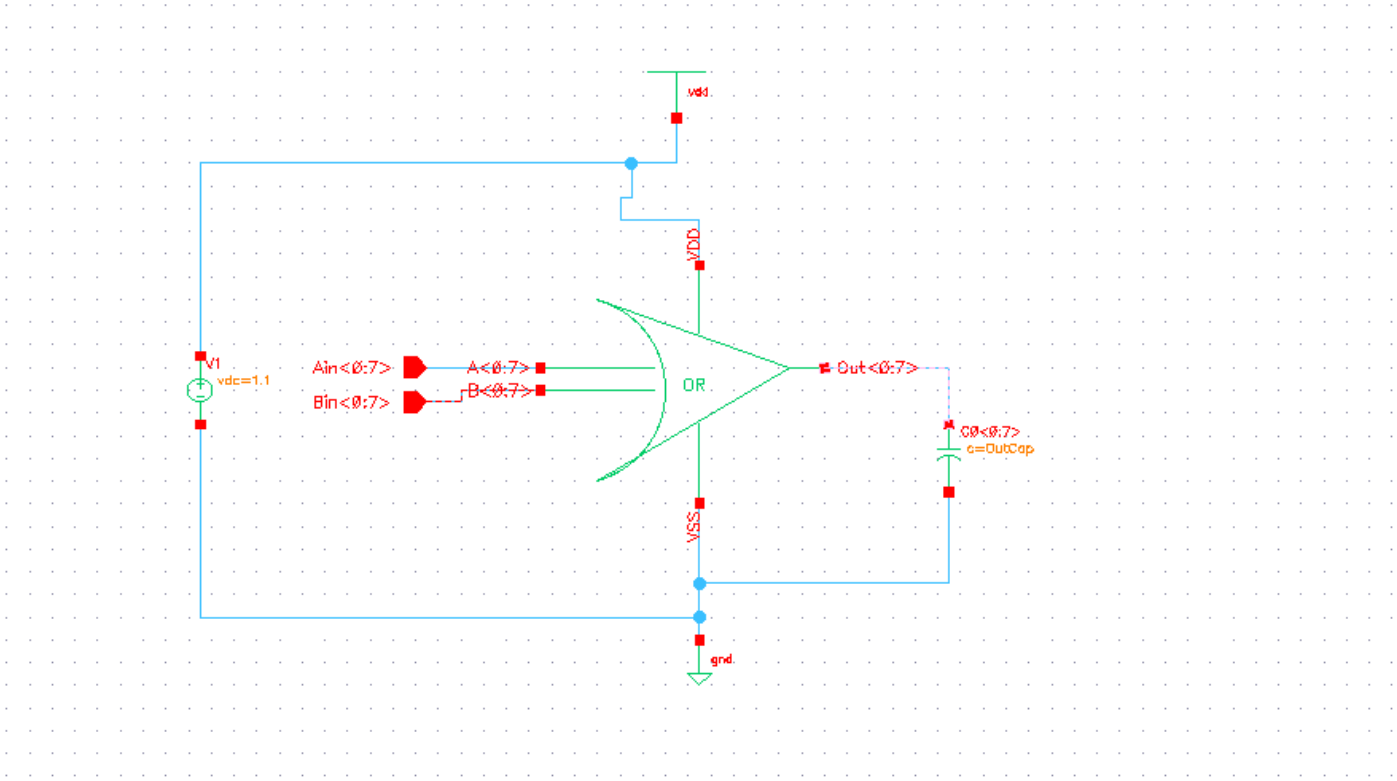

Figure 3: Schematic View of the Test Bench for the Or Gate

To test the circuit for the Or gate, a test bench was first constructed using the cell view in Cadence. Then using spectre the value of the input A was set to low and the value of the input B was swept from 0 V to 1.1 V. If the output switches from low (0 or 0) to high (0 or 1.1) then the circuit functions as an Or gate. The results of the simulation is plotted below:

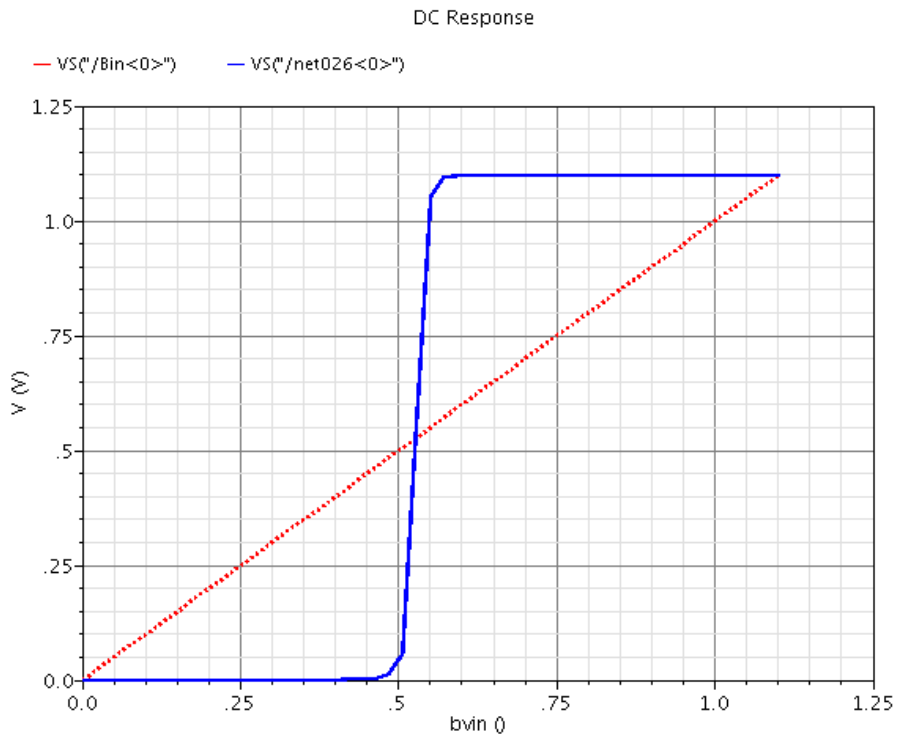

Figure 4: Simulation results with the red dotted line as the input B and the output being the Blue line

From Figure 4 it can be seen that the circuit behaves as an Or gate.

Pass gate (transmission gate)

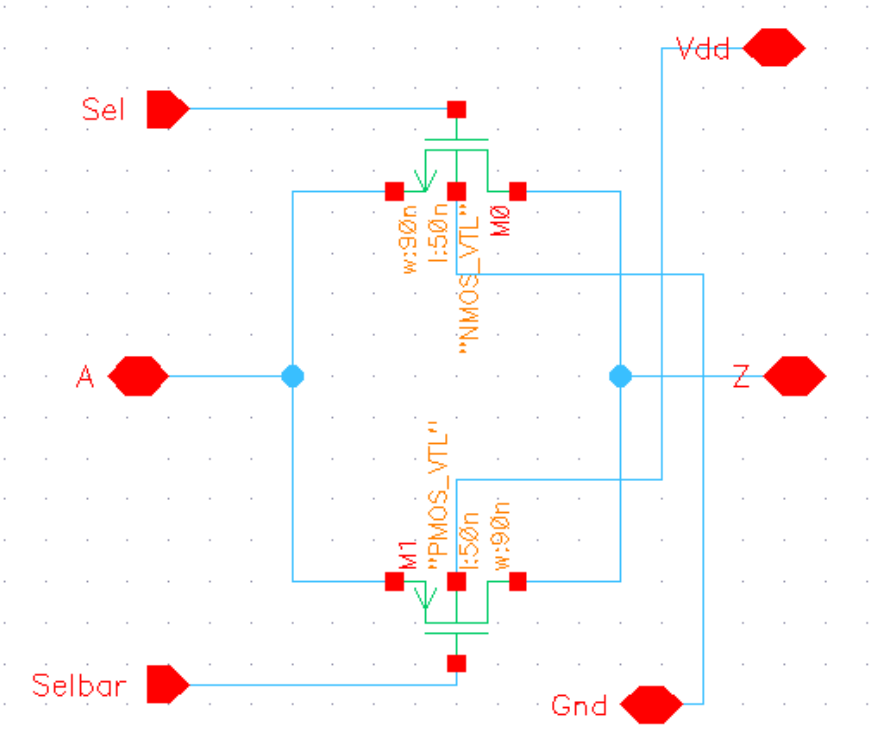

Figure 5: Schematic View of the transmission gate, or pass gate

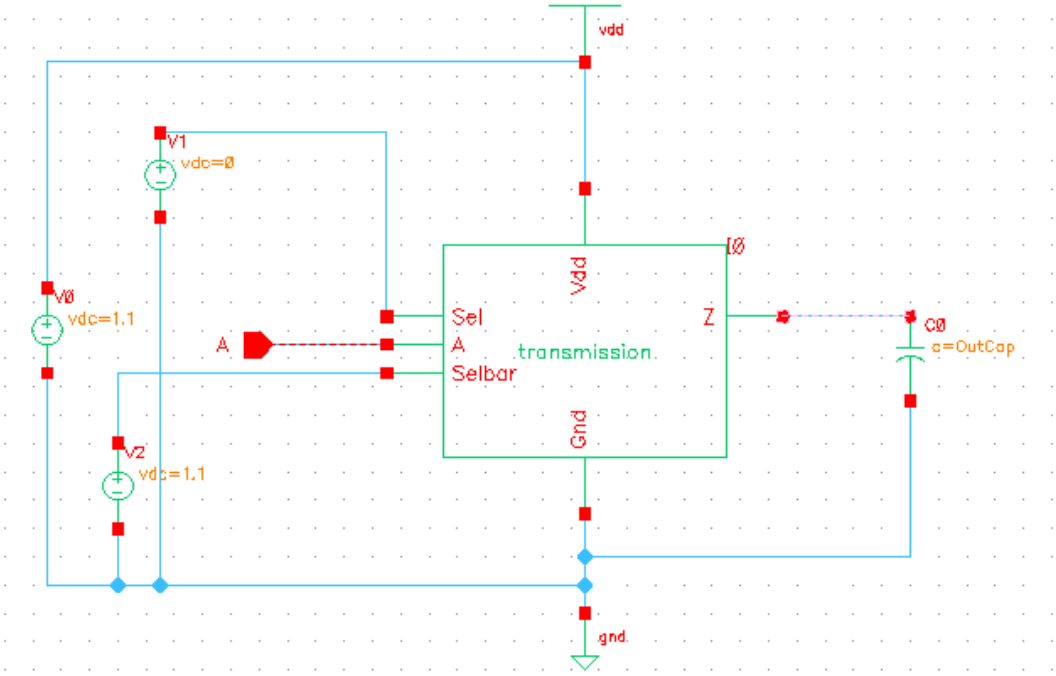

Figure 6: Schematic View of the test bench transmission gate, or pass gate

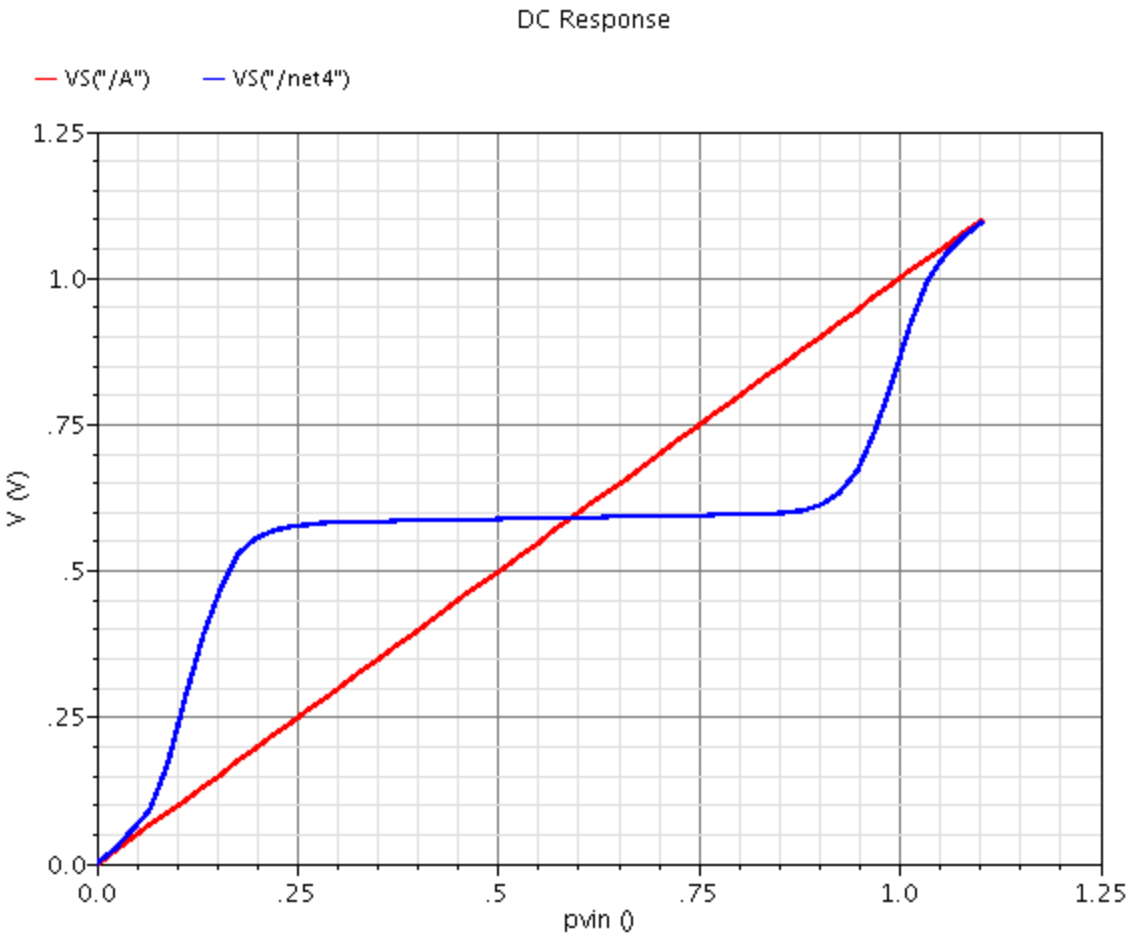

Figure 7: Simulation result of pass gate with blue as output and red as input

2:1 Mux

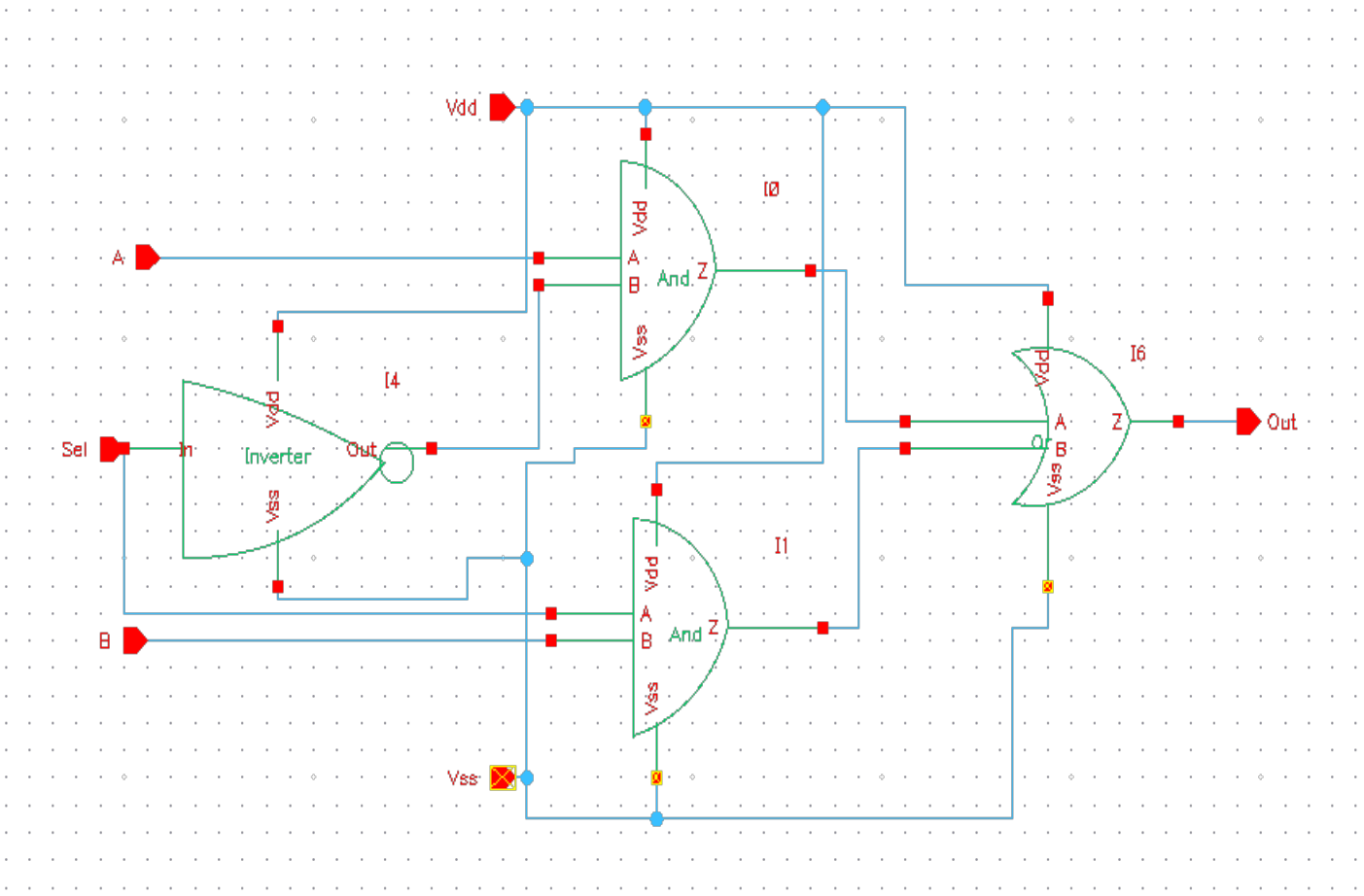

Figure 8: Schematic View of the mux

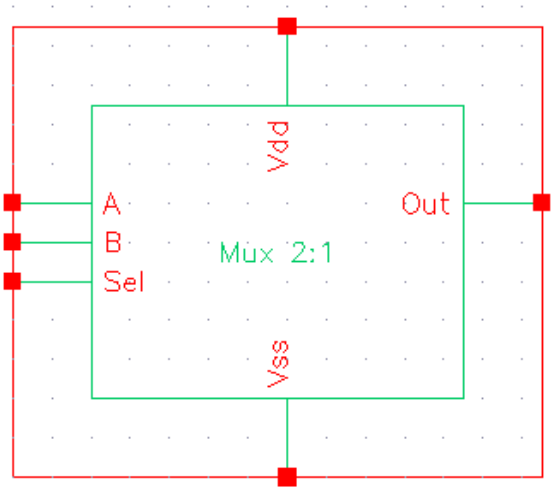

Figure 9: Symbol View of the mux

We are unable to get the mux to simulate unfortunately. Spectre crashes no matter what we try, either DC or trans simulation.

8-Bit And

The And gate was made similarly to the Or gate. A Nand gate was constructed and then an inverter was placed sequentially to realize an And gate.

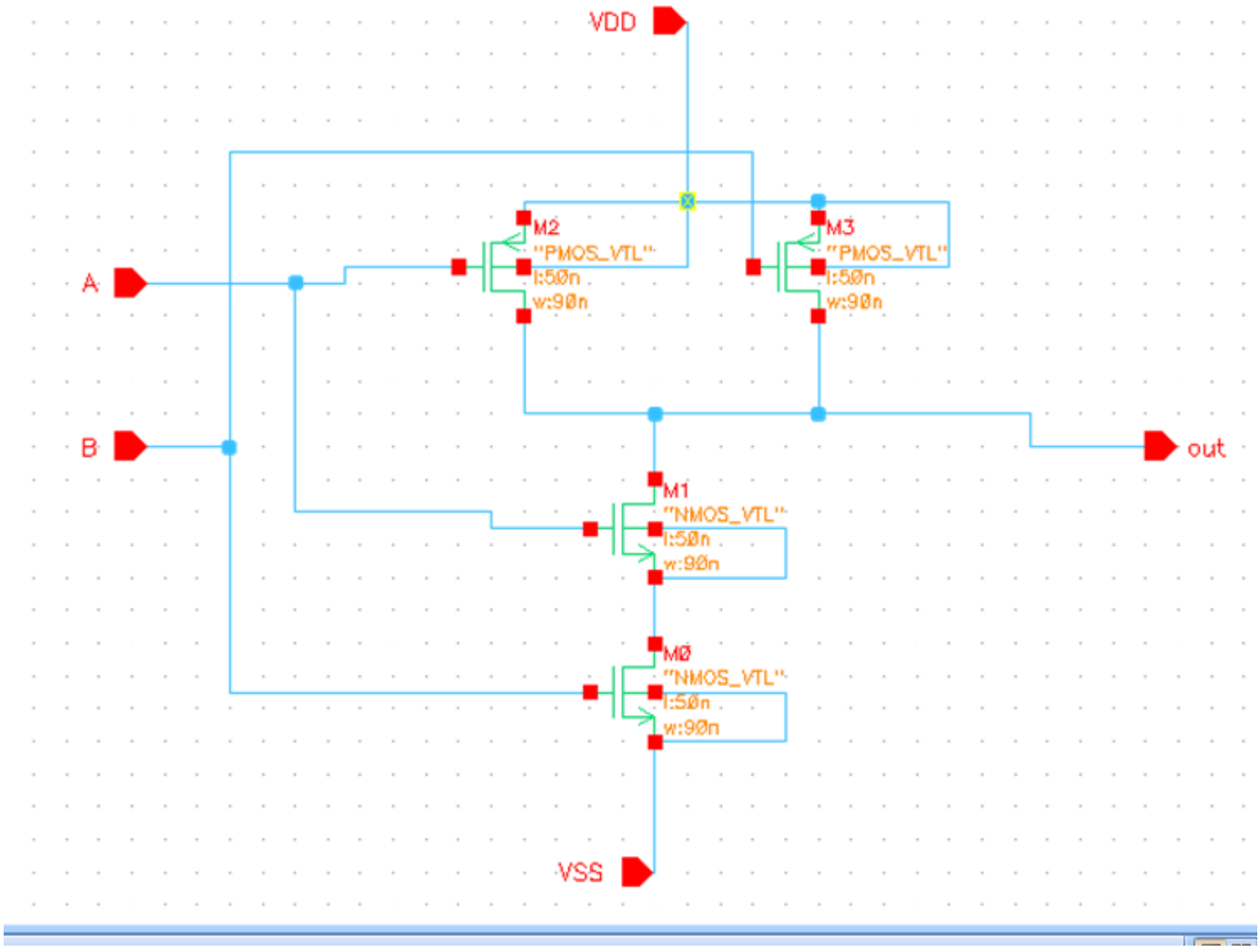

Figure 10: Nand Schematic

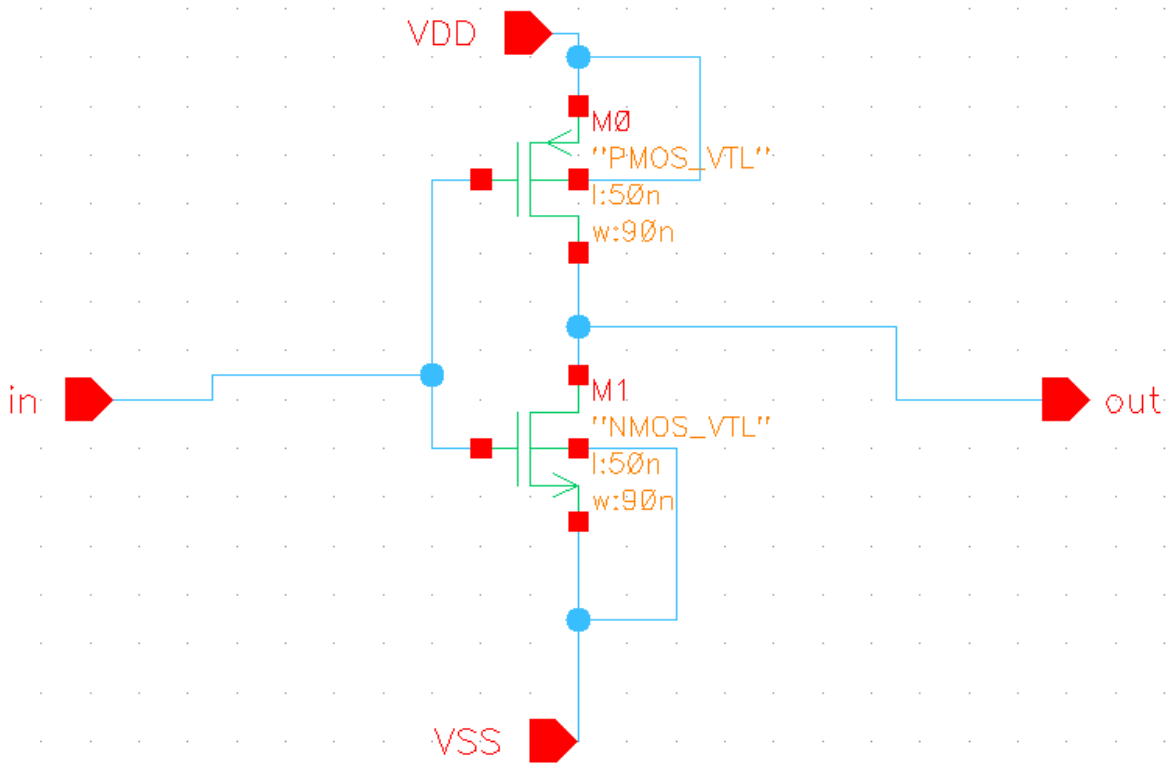

Figure 11: Inverter Schematic

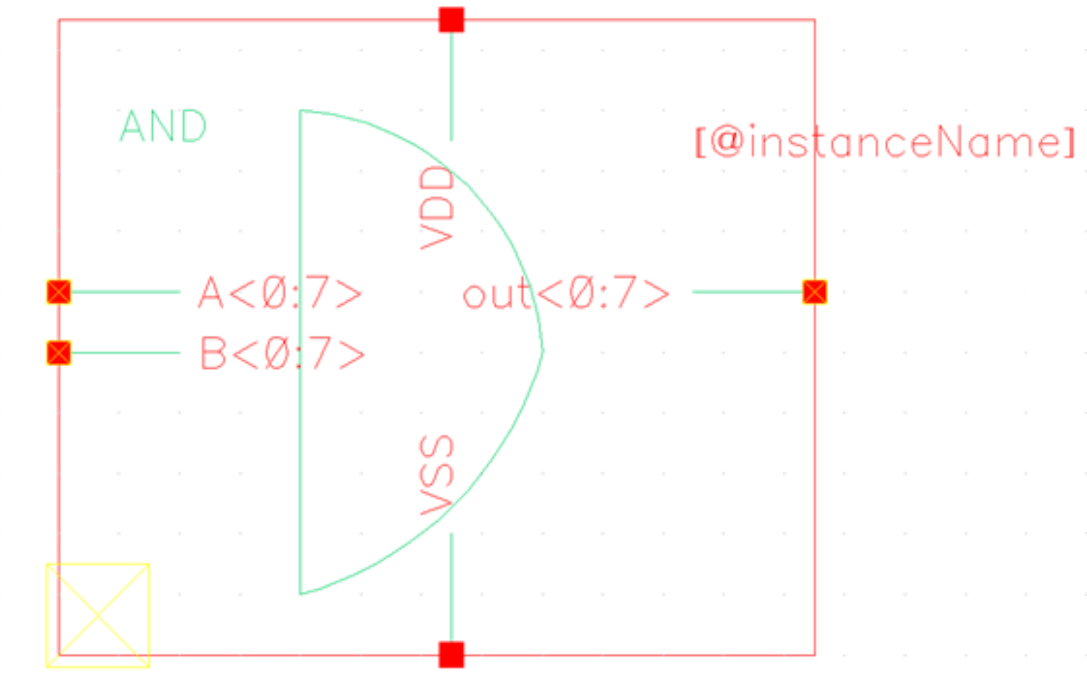

Figure 12: And Symbol

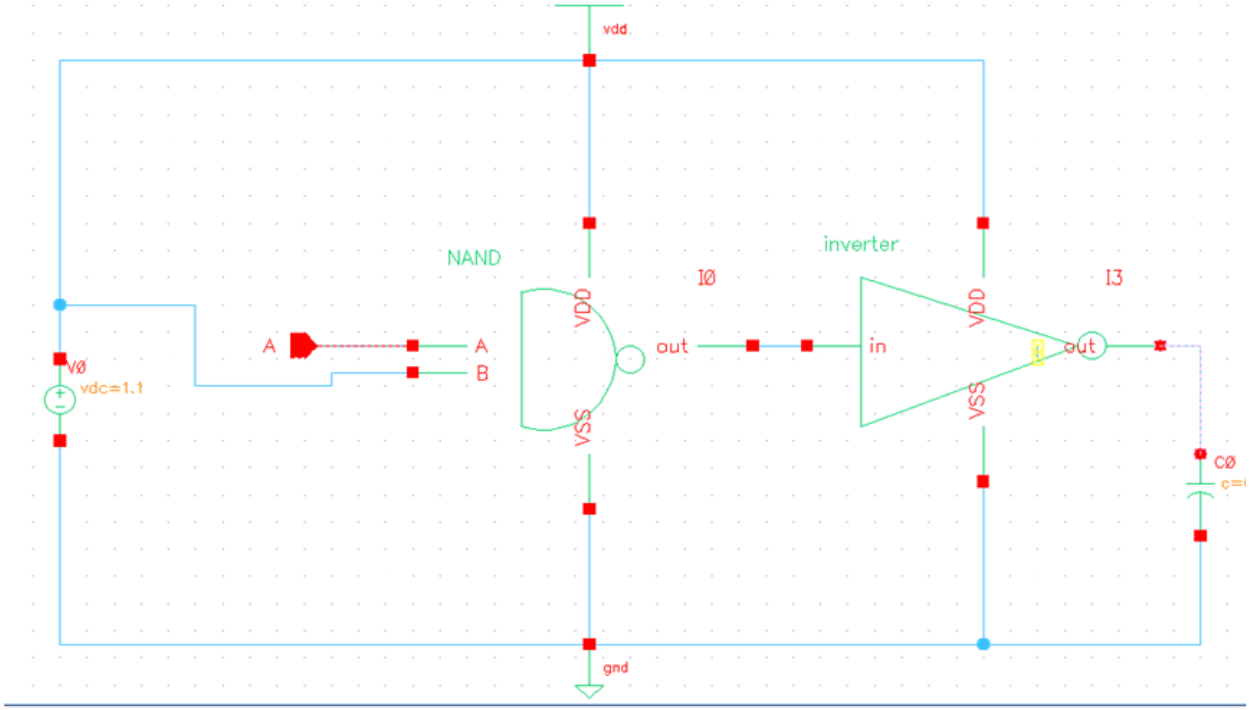

Figure 13: And Test Bench

Report

Our group divided tasks by assigning different gates to different members. We have not been able to meet as a group frequently due to each member's personal reasons and we will try to meet as a group more often in the next report. As of now, we have the schematics and symbols for the AND, OR, PASS, and MUX gates. We also have successful simulation of the AND, OR, and PASS gates but we are having trouble with the MUX simulation (It crashes no matter what we try). For next report, we need to do ADD, 2COMP, SHIFT, ALU gates and figure out how we are going to put everything in the ALU. We plan to divide up the schematics and finish them by week 1. Then meet as a group and work on simulations and integration by week 2.

Our idea of an arbitrary function will be the: XOR, MUL or DIV gates. Our plan after we implement our top design - including all registers, ALU and Inputs/Outputs - we will have a testing strategy in place to ensure all components work together according to specifications. Our testing strategy will consist of having a test bench of known components that work that are included in the Cadence Libraries and testing those components in parallel with our design and seeing if the outputs match for the same inputs (this will span over a wide array of inputs). We will have results outputted to a document where we can analyse the results and see if there are any discrepancies which we need to fix.